

AMENDMENTS TO THE CLAIMS

1. (Currently amended) An information processing system, comprising:

a first computing device comprising:

a first physical port; and

a second physical port;

[[and]] wherein the first computing device is configured to autonomously, without invoking services of a host operating system:

receive, from a first network coupled to the first port, an initialization packet originating from a client;

output a response packet to the client;

receive, from the first network coupled to the first port, a request packet originating from the client; and

based on at least a state of the first computing device and a second computing device, selectively output the request packet to the second computing device, wherein request packets selectively output by the first computing device are output through a second network logically separate from the first network and coupled to the second port.

2. (Currently amended) The system of Claim 1 wherein the first computing device ~~comprises~~ is a network interface card.

3. (Previously presented) The system of Claim 1 wherein the second computing device is configured to perform an operation of a software application in response to receipt of a request packet.

4. (Previously presented) The system of Claim 3 wherein the software application is a socket-based application.

5. (Previously presented) The system of Claim 1 wherein the initialization packet is addressed by the client to the first computing device, and wherein the first computing device is configured to receive the initialization packet in response to the addressing.

6. (Previously presented) The system of Claim 3 wherein the operation includes outputting a response packet to the client, and wherein the first computing device is configured to:

in response to at least the request packet and the state, selectively output the request packet to the second computing device for outputting the response packet to the client, such that the output response packet bypasses the first computing device.

7. (Previously presented) The system of Claim 1 wherein the first network comprises a global computer network.

8. (Previously presented) The system of Claim 7 wherein the second network comprises a local area network.

9. (Currently amended) The system of Claim 1 wherein the first computing device is configured to autonomously, without invoking services of a host operating system:

in response to at least the initialization packet, establish a data structure associated with a connection with the client; and

in response to at least the request packet and the state, selectively output a reference to the data structure to the second computing device for associating an application of the second computing device with the connection.

10. (Currently amended) The system of Claim 1 wherein the first computing device is configured to autonomously, without invoking services of a host operating system:

in response to at least the initialization packet, establish a data structure associated with a connection with the client, the data structure including a group of sequence numbers associated with the connection.

11. (Currently amended) The system of Claim 10 wherein the first computing device is configured to autonomously, without invoking services of a host operating system:

in response to at least the request packet and the state, selectively output a reference to the data structure to the second computing device for performing an operation in response to receiving the reference to the data structure, the operation including outputting a response packet to the client according to the group of sequence numbers, such that the output response packet bypasses the first computing device and appears to the client as received from the first computing device.

12. (Original) The system of Claim 10 wherein the group of sequence numbers includes at least one start sequence number, at least one current sequence number, and at least one acknowledgement sequence number.

13. (Currently amended) The system of Claim 1 wherein the first computing device is configured to autonomously, without invoking services of a host operating system:

in response to at least the initialization packet, establish a data structure associated with a connection to the client, the data structure including an address of the first computing device; and

in response to at least the request packet and the state, selectively output a reference to the data structure to the second computing device for performing an operation in response to receiving the reference to the data structure, the operation including outputting a response packet

to the client with a source address that indicates the address of the first computing device, such that the output response packet bypasses the first computing device and appears to the client as received from the first computing device.

14. (Original) The system of Claim 13 wherein the address includes an IP address.

15. (Original) The system of Claim 14 wherein the address includes a port.

16. (Original) The system of Claim 15 wherein the port is a TCP port.

17. (Original) The system of Claim 15 wherein the port is a UDP port.

18. (Currently amended) A method performed by a first computing device of an information processing system, the first computing device coupled to a first network via a first physical port and coupled to a second network via a physical second port, the first and second networks being logically separate, comprising:

receiving, through the first port, an initialization packet originating from a client;

outputting a response packet to the client;

receiving, through the first port, a request packet originating from the client; and

based on at least a state of the first computing device and a second computing device, selectively outputting the request packet to the second computing device for performing an operation in response to the request packet, wherein request packets selectively output by the first computing device are output through the second port,

wherein the method is performed by the first computing device autonomously and without invoking services of a host operating system.

19. (Currently amended) The method of Claim 18 wherein the first computing device ~~comprises~~ is a network interface card.

20. (Original) The method of Claim 18 wherein the operation is part of a software application.

21. (Previously presented) The method of Claim 20 wherein the software application is a socket-based application.

22. (Previously presented) The method of Claim 18 wherein the initialization packet is addressed by the client to the first computing device, and wherein the receiving, through the first port, an initialization packet originating from a client comprises:

receiving the initialization packet in response to the addressing.

23. (Previously presented) The method of Claim 18, the selectively outputting, through the second port, the request packet to the second computing device for performing the operation in response to the request packet comprising:

selectively outputting, through the second port, the request packet to the second computing device, for outputting a response packet to the client that bypasses the first computing device.

24. (Previously presented) The method of Claim 18, the first network comprising a global computer network, and further comprising:

receiving the initialization packet through the global computer network.

25. (Previously presented) The method of Claim 24, the second network comprising a local area network, and further comprising:

selectively outputting the request packet to the second computing device through the local area network.

26. (Previously presented) The method of Claim 18, comprising:

in response to at least the initialization packet, establishing a data structure that represents a connection with the client; and

in response to at least the request packet and the state, selectively outputting a reference to the data structure to the second computing device for associating an application of the second computing device with the connection.

27. (Previously presented) The method of Claim 18, comprising:

in response to at least the initialization packet, establishing a data structure that represents a connection with the client, the data structure including a group of sequence numbers associated with the connection.

28. (Previously presented) The method of Claim 27, further comprising:

in response to at least the request packet and the state, selectively outputting a reference to the data structure to the second computing device for outputting a response packet to the client according to the group of sequence numbers, such that the output response packet bypasses the first computing device.

29. (Original) The method of Claim 27 wherein the group of sequence numbers includes at least one start sequence number, at least one current sequence number, and at least one acknowledgement sequence number.

30. (Previously presented) The method of Claim 18, further comprising:

in response to at least the initialization packet, establishing a data structure that represents a connection with the client, the data structure including an address of the first computing device.

31. (Original) The method of Claim 30 wherein the address includes an IP address.

32. (Original) The method of Claim 31 wherein the address includes a port.

33. (Original) The method of Claim 32 wherein the port is a TCP port.

34. (Original) The method of Claim 32 wherein the port is a UDP port.

35. (Currently amended) An intelligent network interface device, comprising:
a first physical port for receiving an IP request packet from a client over a first network;
and

a second physical port for transmitting the received IP request packet over a second network that is logically separate from the first network; and

a processor that is structured to autonomously and without invoking services of a host operating system:

maintain a state information table; and

selectively transmit the received IP request packet to a second intelligent network interface device based at least in part on the state information table, so that the second intelligent network interface device causes a response IP packet to be sent transparently to the client, wherein selectively transmitted request packets transmitted by the processor are transmitted through the second port.

36. (Previously presented) The intelligent network interface device of claim 35, further comprising:

a synchronization port for receiving synchronization information, wherein the processor maintains the state information table based at least in part on the received synchronization information.

37. (Previously presented) The intelligent network interface device of claim 36 wherein the synchronization port comprises a third IP port.

38. (Previously presented) The intelligent network interface device of claim 35, further comprising:

a third port for selectively transmitting a response packet in response to a received request packet over a third network that is logically separate from the first network and the second network.

39. (Currently amended) A server farm, comprising:

a plurality of servers, each server comprising a first computing device;

a first network to receive client requests and coupled to the first computing device of a first server of the plurality of servers;

a second network logically and physically separate from the first network, configured to redistribute received client requests and coupled to the first computing device of each of the plurality of servers; and

a third network logically and physically separate from the first and second networks, configured to transmit synchronization information and coupled to the first computing device of each of the plurality of servers,

wherein the first computing device of the first server of the plurality of servers is configured to autonomously, and without invoking services of a host operating system, selectively redistribute a client request received via the first network to a second server in the

plurality of servers based at least in part on synchronization information transmitted via the third network, wherein redistributed client requests are redistributed through the second network.

40. (Currently amended) The server farm of claim 39, further comprising:
~~a router for transmitting a response to a received client request and coupled to the second server in the plurality of servers~~ a first router, coupled to the first server and configured to transmit client requests to the first server; and
a second router, coupled to the second server and configured to transmit a response to the client from the second server;
wherein the response to the client transmitted by the second server bypasses the first router.

41. (Previously presented) The server farm of claim 39 wherein the first network is coupled to each of the plurality of servers and each server can receive a client request.

42. (Currently amended) A computer-readable memory medium containing instructions for controlling a computer processor in a server among a plurality of servers to selectively load balance and direct network requests among the plurality of servers autonomously, without invoking services of a host operating system, by:
maintaining a state table associated with the plurality of servers; and
when a client request packet is received at a first physical port coupled to a first network, selectively generating, based at least in part on the maintained state table, a first type of network packet for transmission to a client through the first network or a second type of network packet for transmission to another processor residing in a different one of the plurality of servers at a second physical port coupled to a second network that is logically separate from the first network.

43. (Previously presented) The computer-readable memory medium of claim 42, further comprising instructions that control the processor by:

when a network packet of the second type is received at the second port coupled to the second network, selectively generating a third type of network packet for transmission to the client through the first network.

44. (Previously presented) The computer-readable memory medium of claim 42, further comprising instructions that control the processor by:

when a network packet of the second type is received at the second port coupled to the second network, selectively generating a third type of network packet for transmission to the client through a third network that is logically separate from the first network and the second network.

45. (Previously presented) The computer-readable memory medium of claim 42 wherein the second type of network packet contains an encapsulation header that identifies client connection information.

46. (Currently amended) A server for use in a computer system server farm, comprising:

means for monitoring a state of the server farm;

means for receiving packets from a global network;

means for generating, without invoking services of a host operating system, a first type of packet and transmitting the first type of packet via the global network when an initialization packet is received from the global network;

means for selectively generating, without invoking services of a host operating system and based at least in part on the monitored state of the server farm, a second type of packet when a client request packet is received from the global network; and

means for transmitting, without invoking services of a host operating system, the second type of packet to another server in the server farm, wherein the means for transmitting the second type of packet bypasses the means for receiving packets from the global network.

47. (Previously presented) The server of claim 46 wherein the means for receiving packets from a global network comprises a connection to a first local area network and the means for transmitting the second type of packet comprises a connection to a second local area network that is logically separate from the first local area network.

48. (Previously presented) The server of claim 47 wherein the means for monitoring the state of the server farm comprises a connection to a synchronization network that is logically separate from the first local area network.

49. (Currently amended) An information processing system, comprising:
a first computer system; and
a second computer system having a network interface card that is structured to:
receive an initialization packet originating from a client;
output a response packet to the client to set up a connection over a network;
receive a request packet originating from the client directed to the connection; and
based on at least a state of at least one of the first computer system and the second computing system, selectively output ~~the request packet~~ a forwarded-request packet to a network interface card of the first computer system thereby migrating the connection to the first computer system, ~~wherein the network interface card of the second computer system selectively outputs~~

~~the request packet to the network interface card of the first computer system without using IP-IP encapsulation~~ the forwarded-request packet encapsulating the request packet and including both a reference to the connection and a field in a header of the forwarded-request packet indicating that the forwarded-request packet is a forwarded packet.

50. (Previously presented) The information processing system of claim 49 wherein the first computer system is structured to transparently output a response packet to the client in response to the received request packet directed to the connection, such that the client perceives the response packet is received from the second computer system.

51. (Currently amended) The information processing system of claim 49 wherein the network interface card of the second computer system selectively creates and outputs the ~~request~~ forwarded-request packet to the network interface card of the first computer system without invoking services of the operating system.

52. (Previously presented) The information processing system of claim 49 wherein the network interface card of the second computer system selectively outputs the request packet to the network interface card of the first computer system without using TCP splicing techniques.

53. (Previously presented) The method of claim 30, further comprising:
in response to at least the request packet and the state, selectively outputting a reference to the data structure to the second computing device for outputting a response packet to the client with a source address that indicates the address of the first computing device, such that the output response packet bypasses the first computing device.

54. (Currently amended) A server farm, comprising:
a first network configured to receive client packets;

a second network logically and physically separate from the first network and configured to redistribute the received client packets; [[and]]

a third network that bypasses the first network and the second network and is configured to transmit server farm synchronization information; and

a fourth network configured to transmit responses to clients, bypassing portions of the first network.

55. (Previously presented) The server farm of claim 54, further comprising:

a plurality of servers coupled to the second and third networks, wherein at least one of the plurality of servers is configured to maintain a state table based at least in part on the transmitted server farm synchronization information.

56. (Previously presented) The server farm of claim 55 wherein the at least one of the plurality of servers is configured to redistribute a client packet based at least in part on the state table.

57. (Currently amended) A method of servicing client requests, comprising:

receiving a client packet through a first network;

selectively redistributing the received client packet through a second network that is logically and physically separate from the first network;

transmitting synchronization information through a third network that bypasses the first network and the second network; and

selectively migrating a client connection based at least in part on the transmitted synchronization information;

wherein the method is performed autonomously and without invoking services of a host operating system.

58. (Previously presented) The method of claim 57, further comprising:
maintaining a state information table based at least in part on the transmitted synchronization information.
59. (Previously presented) The method of claim 57 wherein the selectively redistributing is based at least in part on the transmitted synchronization information.
60. (Previously presented) The method of claim 57 wherein the client packet is a request packet, further comprising performing an operation in response to the request packet.
61. (Previously presented) The method of Claim 60 wherein the operation is part of a socket-based software application.
62. (Previously presented) The method of Claim 57 wherein the first network comprises a global computer network.
63. (Previously presented) The method of Claim 62 wherein the second network comprises a local area network.
64. (Previously presented) The method of Claim 57 wherein a received client packet is an initialization packet.
65. (Previously presented) The method of Claim 64, further comprising:
establishing a data structure that represents a connection with the client.
66. (Previously presented) The method of Claim 65 wherein the data structure includes a group of sequence numbers associated with the connection.

67. (Previously presented) The method of Claim 66, further comprising:
outputting a response packet to the client according to the group of sequence numbers.

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100